PERFORMANCE OF THE OVERFLOW-MLP CFD CODE ON THE NASA AMES 512 CPU ORIGIN SYSTEM

James R. Taft

NAS Technical Report NAS-00-005 March 2000

NASA Ames Research Center
Moffett Field, CA 94035
jtaft@nas.nasa.gov
(650) 604-0704

ABSTRACT

The shared memory Multi-Level Parallelism (MLP) technique, developed last year at NASA Ames has been very successful in dramatically improving the performance of important NASA CFD codes. This new and very simple parallel programming approach was first inserted into the OVERFLOW production CFD code in FY 1998 (ref 1). MLP is a vastly simplified, and inherently more scalable alternative to MPI, that takes advantage of the new large CPU count shared memory computing platforms. At the conclusion of the 1998 effort, OVERFLOW-MLP performance scaled linearly to 256 CPUs on the NASA Ames 256 CPU Origin 2000 system (steger). Overall performance exceeded 20.1 GFLOP/s, or about 4.5x the performance of a dedicated 16 CPU C90 system. All of this was achieved without major modification to the original vector based code. The OVERFLOW-MLP code is now in production on the inhouse Origin systems as well as being used offsite at commercial aerospace companies.

Partially as a result of this work, NASA has purchased a new 512 CPU Origin 2000 system to further test the limits of parallel performance for NASA codes of interest. This paper presents the performance obtained from the latest optimization efforts on this machine for the OVERFLOW-MLP code.

1.0 OVERFLOW

The OVERFLOW CFD code is extensively used in the government and commercial aerospace communities to evaluate new aircraft designs. It is one of the largest consumers of NASA supercomputing cycles, and large simulations of highly resolved full aircraft are routinely undertaken. Typical large problems might require hundreds of Cray C90 CPU hours to complete.

Last year's dramatic performance gains with OVERFLOW-MLP on the 256 CPU steger system are exciting. The potential for obtaining results in a day instead of months is revolutionizing the way in which aircraft manufacturers are looking at future aircraft simulation work. The new 512 CPU lomax system offers even greater performance potential, with results back in hours.

Perhaps more importantly than just raw performance, large systems like lomax have finally achieved the processing capability to turn aircraft design into a fully interactive task. It is quite possible at this point to envision "flying" a highly resolved aircraft through the "electronic" wind tunnel in which one can vary the angle of attack, etc at will, and watch the effects in quasi-real time. This has been an elusive goal for decades. It appears that it is now about to be realized.

1.1 Optimization Issues

There were a number of parallel scaling issues that arose in moving from 256 to 512 CPUs. None were unexpected, though the extent of their impact was uncertain. There were three major areas of concern. First, the problem size under consideration remained the same, but the CPU count had doubled. With OVERFLOW, this implied that the fine-grain loop level parallelism would have to scale to twice as many CPUs in order to continue the overall linear scaling seen in the past. Historically, this was difficult to achieve with OVERFLOW for CPU counts greater than 16. Second, the layout of data in system memory would have to be watched more carefully, as the potential for data residing very far away in the NUMA sense, was greater in the new system. A significant increase in remote data references can adversely impact scaling. Finally, the new 512 CPU system was subject to operating system scaling issues. There were many concerns that the normal memory allocation, context switching, etc. activities of the OS were not sufficiently parallelized, or optimized for such a large CPU count machine.

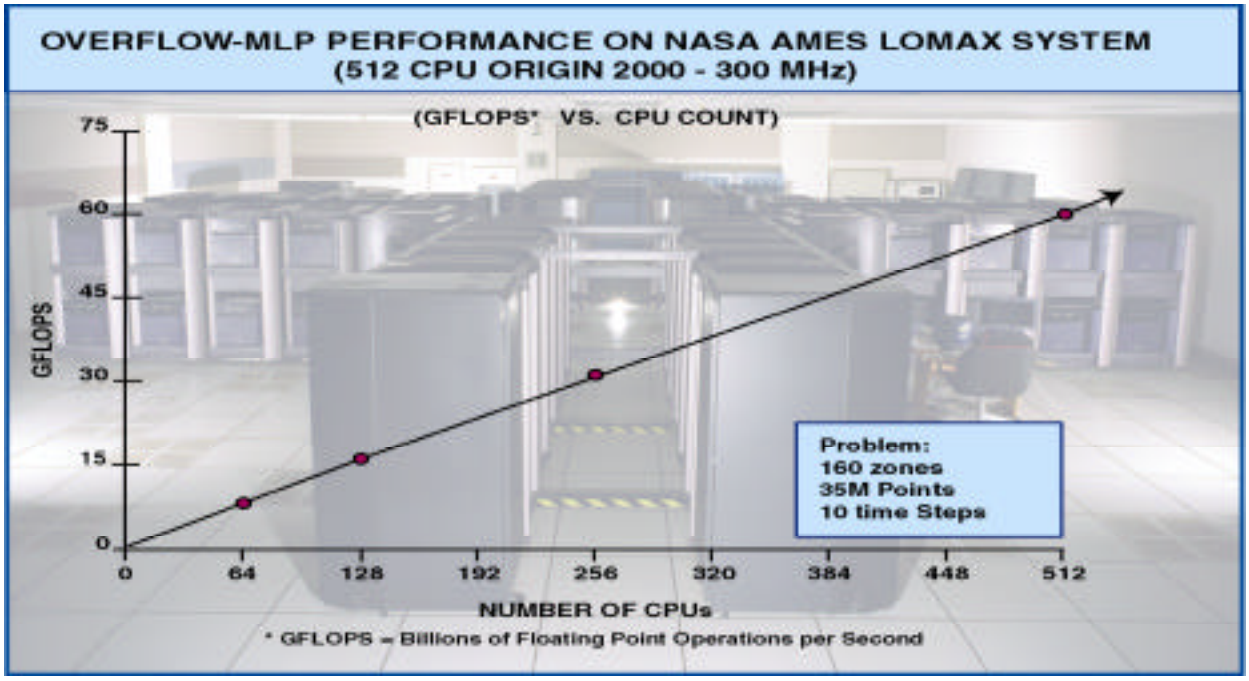
We address the last issue first. The lomax system is a true testbed in that it is the first 512 CPU, shared memory symmetric multi-processing system in the world. It was developed at NASA's request by SGI, and went operational at NASA Ames in October of 1999. The system was intended to test the limits of shared memory parallelism and operating system scalability. The first two weeks of testing revealed a number of system idiosyncrasies that either prevented executions, or dramatically affected application performance. These issues focussed around spawning large numbers of processes, barrier synchronization between processes, and memory addressing/access patterns across the entire system. Within the first two weeks all of these issues were corrected and/or had trivial workarounds that inflicted no performance penalties. All fixes were either resets of system tuning parameters, downloaded system prom values, or user accessible environmental variables.

The principal concern regarding application performance was that the fine grained loop level parallelism was beginning to show the effects of Amdahl's law falloff. That is, loop level parallelism using 16 CPUs was not twice as fast as the same execution on 8 CPUs. A re-evaluation of the loop level parallelism in the code identified a few serial loops and routines that had been missed before. In particular, boundary condition end cases needed to be more carefully addressed, and parallelism exploited. These previously untreated serially executing sections did not substantially contribute to degradation in the 256 CPU runs, but were very noticeable in the 512 CPU tests. The simple fix was to include additional parallel directives in the code. At this point fine grained parallel efficiency is about 90% when going from 16 to 32 CPUs. This is more than adequate for the vast majority of problems envisioned for OVERFLOW.

The final concern was that the larger system would experience larger latency effects in accessing memory across the much larger hypercube interconnect. In the past, the remote access had not been a problem in that data was often fetched from remote locations and copied into local memory where it was then used for a considerable amount of the time. The migration to 512 CPUs required the few remaining widely distributed arrays to follow this local copy concept to maintain computational efficiency.

1.2 Performance Results

Figure 1 below is a plot of current OVERFLOW-MLP performance on the dedicated 512 CPU lomax system (pictured in the background). As can be seen, the chart indicates that OVERFLOW-MLP continues to scale linearly with CPU count up to 512 CPUs on a large 35 million point full aircraft simulation. At this point performance is such that a fully converged simulation of 2500 time steps is completed in less than 2 hours of elapsed time.



The current performance results represent a sustained processing rate of about 117 Cray C90 equivalent MFLOP/s per CPU. This is approximately 20% of the peak performance of the R12000 processor. Summing over 512 CPUs, the aggregate overall performance is about 60 GFLOP/s or 13X that of a dedicated 16 CPU C90 system for the same problem. Though not shown above, the OVERFLOW-MLP performance is also more than 3x the OVERFLOW-MPI performance for problems of this type executing on 512 CPUs

1.3 Future Work

Even though OVERFLOW-MLP performance is very high relative to the C90, current performance levels are substantially less than can be obtained with this code on the new microprocessor based systems. Historically, the entire focus on optimizing OVERFLOW has been to increase the efficiency of fine and coarse-grained parallelism. Virtually no single CPU optimizations have been performed. Examination of the code has shown that there is perhaps a factor of two in runtime reduction still available if this activity is undertaken. This work will be substantially more involved than the work to date, as many routines will change and the verification and validation effort will expand accordingly. Current plans are to move forward with this effort as programmer resources become available.

2.0 Summary

The recent addition of the single system 512 CPU Origin to the NASA Ames NAS facility has proven to be highly successful. Researchers are now capable of executing high fidelity OVERFLOW-MLP simulations of full aircraft at 60 GFLOP/s. Even more importantly, this rate has revolutionized the way in which production CFD for these kinds of problems can be done. High fidelity OVERFLOW simulations can now be accomplished in an hour or two, opening the way to true interactive aircraft design. The long sought goal of the Electronic Wind Tunnel, or "e-tunnel" is now at hand.

The success with OVERFLOW-MLP demonstrates that the MLP technique is an important step forward in improving parallel scaling efficiency for CFD codes important to NASA's continuing missions. The MLP parallel scaling approach will be inserted into a number of codes this year. Currently, it is being distributed to the user community for evaluation and test.

3.0 References

Taft, J.R. Multi-Level Parallelism, A Simple Highly Scalable Approach to Parallelism for CFD, HPCCP/CAS Workshop 98 Proceedings, Catherine Schulbach, editor.